

IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE



Membership Publications/Services Standards Conferences Careers/Jobs

IEEE Xplore
RELEASE 1.0

 Welcome
 United States Patent and Trademark Office


» ABSTRACTS

[Help](#) [FAQ](#) [Terms](#) [IEEE Peer Review](#)

Quick Links

Welcome to IEEE Xplore

- ☐ Home
- ☐ What Can I Access?
- ☐ Log-out

Tables of Contents

- ☐ Journals & Magazines
- ☐ Conference Proceedings
- ☐ Standards

Search:

- ☐ By Author
- ☐ Basic
- ☐ Advanced

Member Services

- ☐ Join IEEE
- ☐ Establish IEEE Web Account
- ☐ Access the IEEE Member Digital Library

[Search Results](#) [\[PDF FULL-TEXT: 54 KB\]](#) [PREV](#) [NEXT](#) [DOWNLOAD CITATION](#)

 Request Permissions
RIGHTSLINK

Dynamic hammock predication for non-predicated instruction set architectures

[Klauser, A.](#) [Austin, T.](#) [Grunwald, D.](#) [Calder, B.](#)

Dept. of Comput. Sci., Colorado Univ., Boulder, CO , USA;

This paper appears in: Parallel Architectures and Compilation Technique Proceedings. 1998 International Conference on

Meeting Date: 10/12/1998 - 10/18/1998

Publication Date: 12-18 Oct. 1998

Location: Paris France

On page(s): 278 - 285

Reference Cited: 24

Number of Pages: xiii+435

Inspec Accession Number: 6084632

Abstract:

Conventional speculative architectures use branch prediction to evaluate the execution path during program execution. However certain branches are difficult to predict. One solution to this problem is to evaluate both paths following such a conditional branch. Predicated execution can be used to implement this form of path execution. Predicated architectures fetch and issue instructions that have associated predicates. These predicates indicate if the instruction should complete its result. Predicating a branch reduces the number of branches executed, eliminating the chance of branch misprediction at the cost of executing additional instructions. In this paper, we propose a restricted form of multi-path execution called Dynamic Predication for architectures with little or no support for predicated instructions in their instruction set. Dynamic predication dynamically predicates instruction sequences in the form of a branch hammock concurrently executing both paths of a branch. A branch hammock is a short forward branch that spans a few instructions in the form of an if-then or if-then-else construct. We mark these and other constructs as executable. When the decode stage detects such a sequence, it passes a preselected instruction sequence to a dynamically scheduled execution core. Our results show that dynamic predication can accrue speedups of up to 13%.

Index Terms:

[parallel architectures](#) [processor scheduling](#) [Dynamic Predication](#) [branch hammock](#) [prediction](#) [conditional branch](#) [if-then](#) [if-then-else](#) [multi-path execution](#) [speculative](#)